
SWISS FEDERAL INSTITUTE OF TECHNOLOGY LAUSANNE
EIDGENÖSSISCHE TECHNISCHE HOCHSCHULE LAUSANNE
POLITECNICO FEDERALE DI LOSANNA

LSL/EPFL Laboratoire de systèmes logiques
INN-Ecublens CH-1015 LAUSANNE, Switzerland



ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Automate cellulaire autoréplicateur de von Neumann

Jean-Luc Beuchat
Jacques-Olivier Haenni

Mars 2001

Chapitre 1

Prolégomènes

1.1 Brève biographie de John von Neumann

John von Neumann est né à Budapest en 1903. Reconnu très tôt comme un enfant prodige, il divisait de tête deux nombres de huit chiffres ou récitait les colonnes de l'annuaire téléphonique afin d'amuser sa famille.

Il émigra aux Etats-Unis en 1931 et devint professeur de mathématiques à l'université de Princeton. Durant la seconde guerre mondiale, il participa à la réalisation de la bombe atomique, à Los Alamos. Il rejoignit le projet ENIAC¹ en 1944. L'architecture de von Neumann, utilisée de nos jours, naquit de ses travaux dans le domaine de l'informatique. Von Neumann décéda des suites d'un cancer, probablement contracté lors d'expérimentations nucléaires, en 1957.

1.2 Principes fondamentaux

Au début des années cinquante, von Neumann s'intéressait au problème de l'autoréplication. Il étudia les mécanismes permettant à une machine de se dupliquer et définit un **constructeur universel**.

Le constructeur universel M_c^* se révèle capable de fabriquer n'importe quelle structure à partir d'une description de cette dernière et des matériaux adéquats. Afin de réaliser l'autoréplication, il suffit de fournir à M_c^* sa propre description. Ce processus s'avère cependant délicat. Etudions l'exemple de la figure 1.1. Nous disposons initialement d'un constructeur universel et de sa description (figure 1.1c). Au terme du processus d'autoréplication, nous obtenons le système illustré par la figure 1.1d et constatons qu'il diffère de l'automate initial. Ne possédant pas sa description, il ne peut se reproduire.

Etudions maintenant l'automate de la figure 1.1b. Outre sa description, il possède une description de sa description. Lors de l'autoréplication, cette dernière permet l'élaboration de la description de l'automate 1.1c.

Nous sommes ainsi confrontés à un problème de régression infinie ! Von Neumann suggéra la solution suivante afin de résoudre cette difficulté :

¹*Electronic Numerical Integrator And Calculator*

- dans un premier temps, la description se comporte comme un génome : elle est interprétée afin de réaliser une copie du constructeur universel;
- elle est ensuite littéralement copiée (figure 1.3).

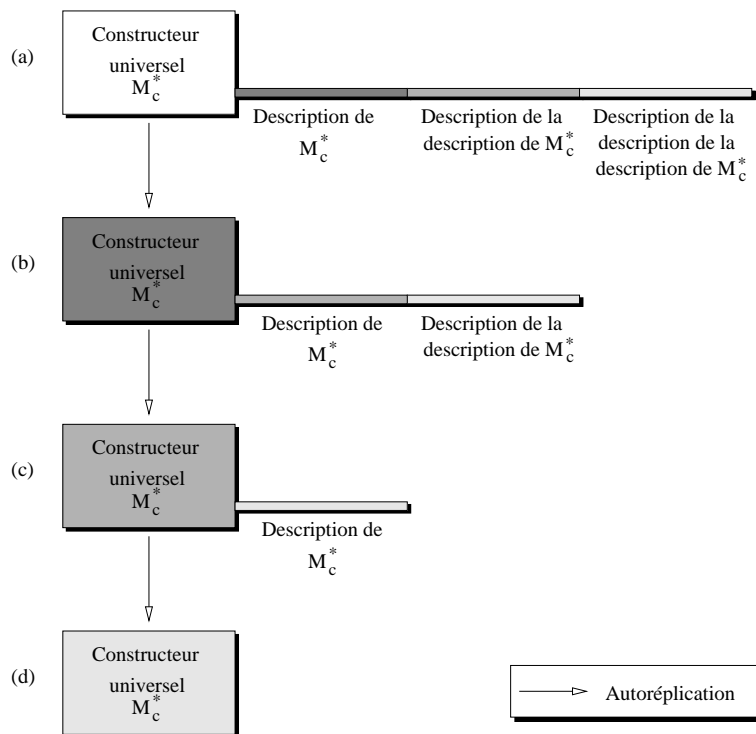


Figure 1.1: *Le problème de régression infinie.*

Stanislas Ulam, un collègue de von Neumann à Los Alamos, avait une passion. Il créait des jeux mathématiques qui généraient d'étranges objets géométriques sur l'écran de son ordinateur. Il divisait ce dernier en une matrice de points qu'il allumait ou éteignait (automate cellulaire). Ulam suggéra à von Neumann l'utilisation d'un tel univers abstrait.

Après plusieurs essais, von Neumann définit un automate cellulaire à 29 états [5]. Il démontra ensuite qu'une certaine configuration de son automate réalisait un constructeur universel possédant les propriétés suivantes :

❶ **Construction de n'importe quel automate à partir de sa description.**

Le constructeur universel se révèle capable de construire n'importe quel automate à partir de sa description. Cette dernière figure sur une bande lue par le constructeur (figure 1.2).

❷ **Autoréplication du constructeur universel** (figure 1.3).

❸ **Autoréplication d'un calculateur universel.**

La figure 1.4 illustre un calculateur universel M_u couplé au constructeur universel M_c^* . Le ruban contient la description des deux automates.

Le constructeur universel se compose

- d'un ruban de taille infinie (comme dans une machine de Turing) contenant les informations nécessaires à la construction d'un automate (description et position dans le réseau cellulaire);
- du constructeur qui lit l'information contenue sur le ruban et élabore le nouvel automate à l'aide d'un bras constructeur.

Remarquons finalement que seules des configurations se reproduisent. Le support physique de l'automate (biodules, mémoire d'un ordinateur) ainsi que l'énergie nécessaire sont donnés à priori.

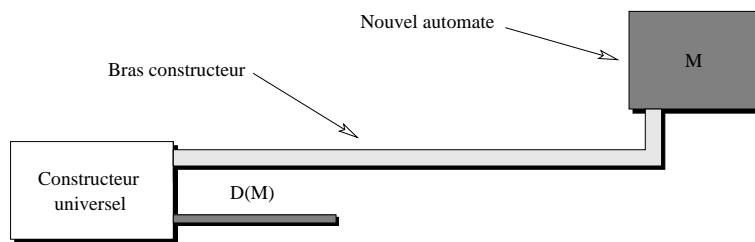


Figure 1.2: *Constructeur universel.* $D(M)$ désigne la description de l'automate M que nous souhaitons construire.

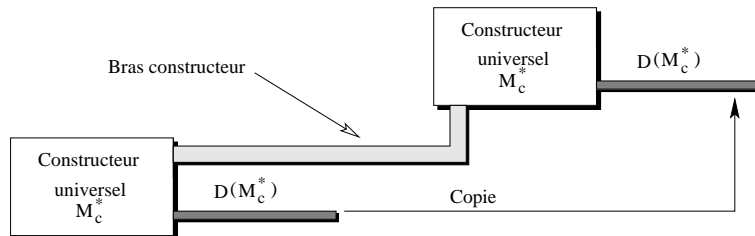


Figure 1.3: *Autoréplication du constructeur universel.*

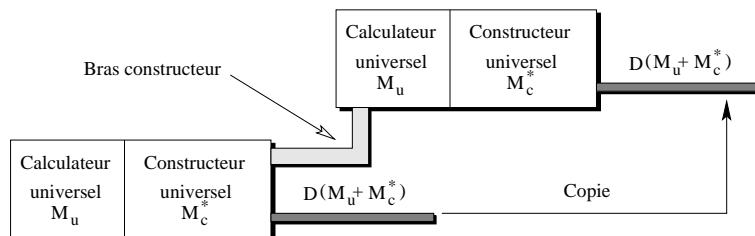


Figure 1.4: *Autoréplication d'une machine de Turing.*

Chapitre 2

Description des états et de la règle de transition

Chaque cellule de l'automate est une machine d'états finie (29 états). Au lieu de numéroter les différents états, il est préférable d'utiliser des symboles graphiques évoquant leur fonctionnalité (tableau 2.1). Nous consacrons ce chapitre à la description de ces différents états et de la règle de transition régissant l'évolution de l'automate.

Etats	Symboles
Etat quiescent	U
Etats sensitifs	$S_\theta, S_0, S_1, S_{00}, S_{01}, S_{10}, S_{11}$ et S_{000}
Etats de transmission ordinaires non excités	$\uparrow, \downarrow, \leftarrow$ et \rightarrow
Etats de transmission ordinaires excités	$\uparrow\cdot, \downarrow\cdot, \leftarrow\cdot$ et $\rightarrow\cdot$
Etats de transmission spéciaux non excités	$\Uparrow, \Downarrow, \Leftarrow$ et \Rightarrow
Etats de transmission spéciaux excités	$\Uparrow\cdot, \Downarrow\cdot, \Leftarrow\cdot$ et $\Rightarrow\cdot$
Confluent non excité	C_{00}
Confluents excités	C_{01}, C_{10} et C_{11}

Tableau 2.1: *Les 29 états.*

2.1 L'état quiescent

Cet état, noté U, correspond à une cellule morte, n'ayant aucune influence sur ses voisines.

2.2 Les états de transmission ordinaires

Les états de transmission ordinaires permettent l'acheminement d'information (excitations) d'un point de l'automate à un autre. Ils sont symbolisés par une flèche indiquant la direction dans laquelle ils transmettent les excitations (nord,

sud, est ou ouest). La face vers laquelle pointe la flèche constitue la sortie de la cellule (les trois autres faces sont les entrées). Un tel état est soit :

- excité; nous symboliserons l'excitation par un point à côté de la flèche;
- non excité.

Il existe ainsi huit états de transmission ordinaires.

Lorsqu'il reçoit une excitation ordinaire (ou provenant d'un confluent) sur une ou plusieurs de ses entrées, un état de transmission ordinaire devient (ou reste) excité. Un tel état se comporte ainsi comme une porte logique OU (figure 2.1).

Un état de transmission ordinaire devient (ou reste) excité lorsqu'il reçoit, sur au moins une de ses entrées, une excitation provenant

- d'un état de transmission ordinaire ou
- d'un état confluent.

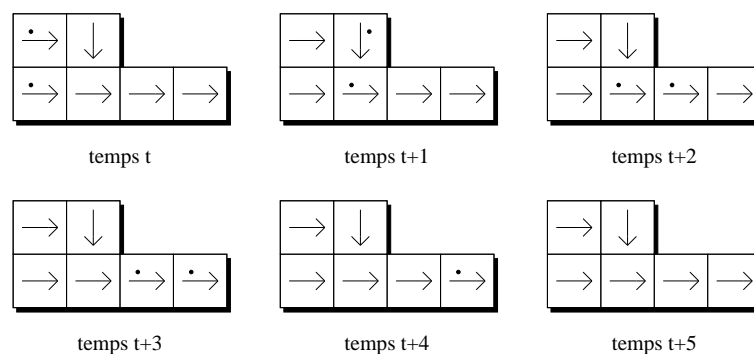


Figure 2.1: *Etats de transmission ordinaires.*

2.3 Les états de transmission spéciaux

Tout comme les états de transmission ordinaires, les états de transmission spéciaux permettent l'acheminement d'information¹ entre deux points de l'automate et se comportent comme une porte logique OU. Une cellule spéciale devient (ou reste) excitée à l'instant $t + 1$ si elle reçoit une excitation provenant d'un état de transmission spécial ou d'un confluent sur une ou plusieurs de ses entrées à l'instant t .

2.4 Les états confluents

Les états confluents $C_{b_1 b_0}$ ($b_1, b_0 \in \{0, 1\}$) remplissent quatre fonctions :

¹Nous constaterons par la suite que cette information diffère de celle transitant à travers des états de transmission ordinaires.

❶ Introduction d'un double délai.

Un état confluent se comporte comme un registre à décalage de deux bits :

- $b_1(t+1) = b_0(t)$
- $b_0(t+1)$ dépend de l'état des cellules voisines au temps t

L'état d'excitation vu par les cellules voisines est $b_1(t)$.

❷ Réalisation de la fonction logique ET.

$b_0(t+1) = 1$ si et seulement si

- il y a au moins une cellule voisine dans un état de transmission ordinaire excité dirigé vers $C_{b_1 b_0}$ et
- toutes les cellules voisines dans un état de transmission ordinaire dirigé vers $C_{b_1 b_0}$ sont excitées au temps t .

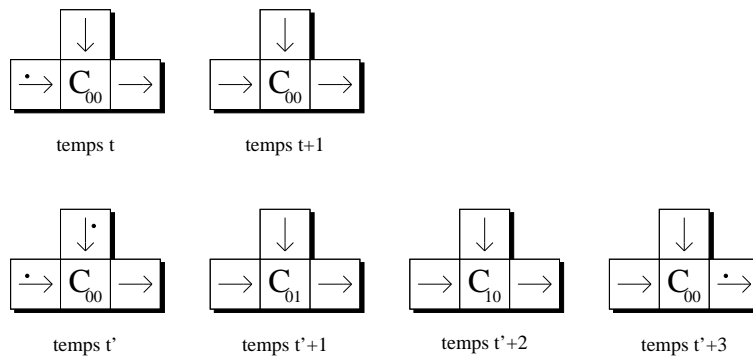


Figure 2.2: Les états confluents se comportent comme une porte logique ET.

❸ Conversion d'une excitation ordinaire en excitation spéciale.

Bien qu'elle ne puisse être excitée que par des états de transmission ordinaires, une cellule $C_{b_1 b_0}$ transmet son excitation à des états de transmission ordinaires ou spéciaux.

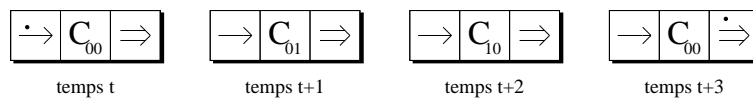


Figure 2.3: Conversion d'une excitation ordinaire en une excitation spéciale à l'aide d'un confluent.

❹ Fan-out.

Un état confluent émet de l'information dans quatre directions (nord, sud, est et ouest). Il est ainsi possible de multiplier un signal en connectant plusieurs états de transmission ordinaires ou spéciaux à un confluent (figure 2.4).

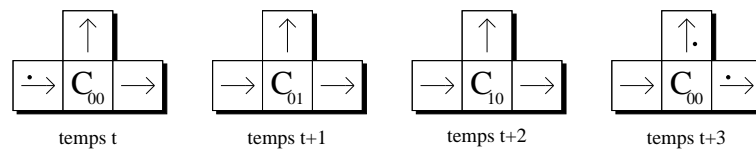


Figure 2.4: Un état confluent se comporte comme un fan-out.

2.5 Les états sensitifs

Les huit états sensitifs sont des états transitoires permettant le passage de l'état U à \leftarrow , \rightarrow , \uparrow , \downarrow , \Leftarrow , \Rightarrow , \Uparrow , \Downarrow ou C_{00} . Lorsqu'un état quiescent reçoit une excitation ordinaire ou spéciale, il passe dans l'état S_θ . A chaque coup d'horloge, l'état évolue selon le graphe de la figure 2.5.

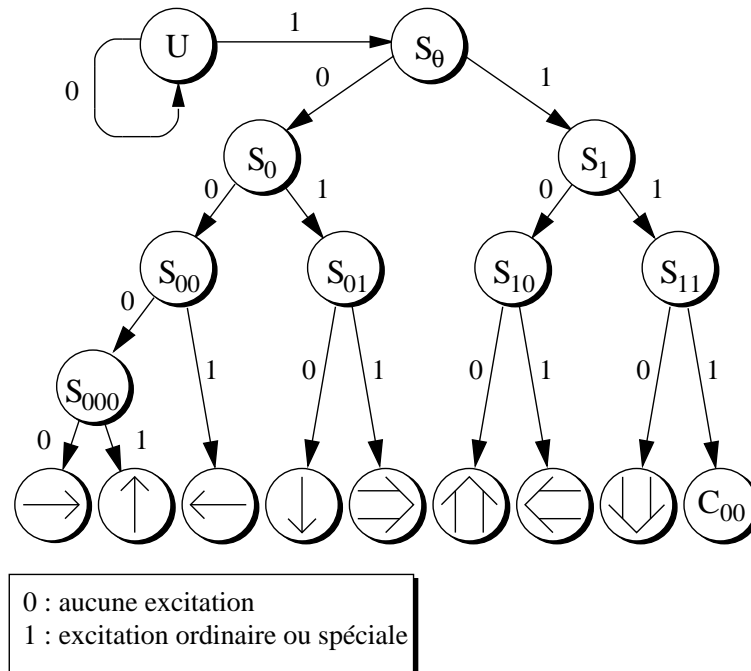


Figure 2.5: Les états sensitifs.

2.6 Destruction d'une cellule

La destruction constitue une transition d'un état de transmission ou confluent vers l'état quiescent. Ce processus survient

- lorsqu'un état de transmission ordinaire ou un confluent reçoit une excitation spéciale sur l'un de ses côtés ou
- lorsqu'un état de transmission spécial reçoit une excitation ordinaire sur l'un de ses côtés.

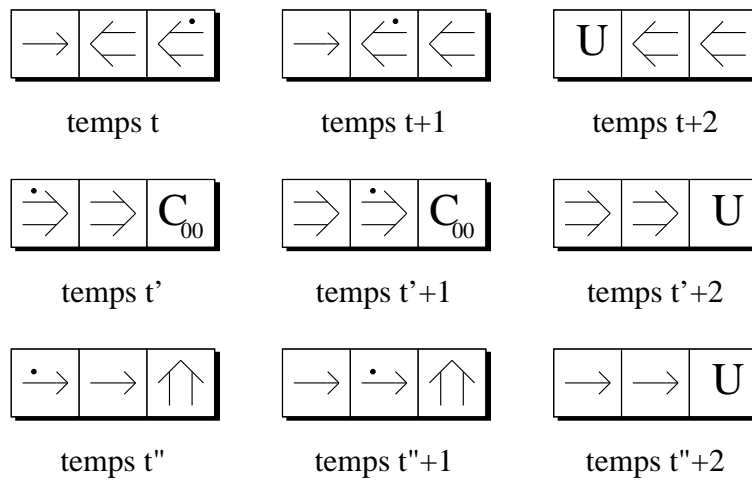


Figure 2.6: *Exemples de destruction de cellules.*

Chapitre 3

Applications

Divers blocs fonctionnels, appelés organes, constituent le constructeur universel. Ce dernier comporte, selon certaines estimations, plus de 200'000 cellules. Nous ne présentons ici que quelques types d'organes.

3.1 Le pulseur

Un pulseur permet de générer des séquences. Une excitation présentée à son entrée produit la séquence $\overline{i_1 i_2 \dots i_n}$ en sortie. Le principe consiste à multiplier l'excitation fournie en entrée grâce à des états confluents, puis d'acheminer ces divers signaux par des chemins de différentes longueurs vers la sortie.

Etudions l'exemple de la figure 3.1. Si l'entrée A reçoit une excitation ordinaire au temps t , nous observons une excitation ordinaire à la sortie B aux instants $t + 11$, $t + 13$ et $t + 16$. Nous obtenons ainsi la séquence de trois excitations 101001.

Le principe de construction d'un pulseur est relativement simple. Soient k et u le nombre de 1 et de 0 apparaissant dans la séquence. La largeur et la hauteur du dispositif sont respectivement de $2 \cdot k$ et $u + 2$ cellules. Remarquons finalement que si la séquence ne comporte qu'un bit égal à zéro, $u = 2$.

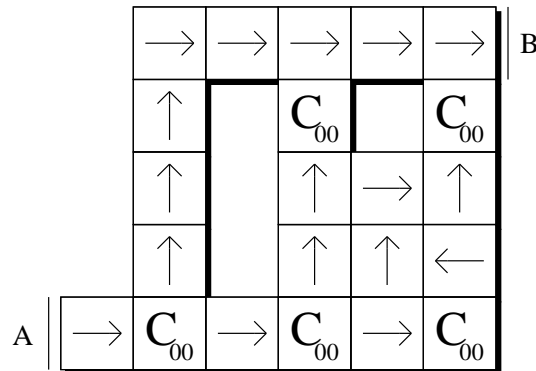


Figure 3.1: Pulseur $P(101001)$.

3.2 Le décodeur

Le décodeur permet de détecter une séquence particulière présentée à son entrée. Soient

- $\overline{j_1 j_2 \dots j_n}$ la séquence d'entrée;
- $\overline{i_1 i_2 \dots i_n}$ la séquence décodée par le dispositif;
- $x_1 x_2 \dots x_m$ la liste des indices des signaux excités de $\overline{i_1 i_2 \dots i_n}$, avec $m \leq n$.

La sortie du décodeur est égale à 1 si

$$j_{x_1} \wedge j_{x_2} \wedge \dots \wedge j_{x_m} = 1$$

La figure 3.2 illustre un décodeur D(1001001). La sortie de ce dernier est donc égale à 1 si

$$j_1 \wedge j_4 \wedge j_7 = 1$$

Soit une séquence d'entrée de longueur n comportant k excitations. La largeur et la hauteur de l'automate sont définies respectivement par $2 \cdot k$ et $n+2$. La difficulté consiste alors à synchroniser les signaux transitant dans le système. Nous calculons d'abord $j_{x_1} \wedge j_{x_2}$ à l'aide d'un confluent. Nous itérons ce processus et déterminons successivement $j_{x_1} \wedge j_{x_2} \wedge j_{x_3}, \dots, j_{x_1} \wedge j_{x_2} \wedge j_{x_3} \wedge \dots \wedge j_{x_m}$.

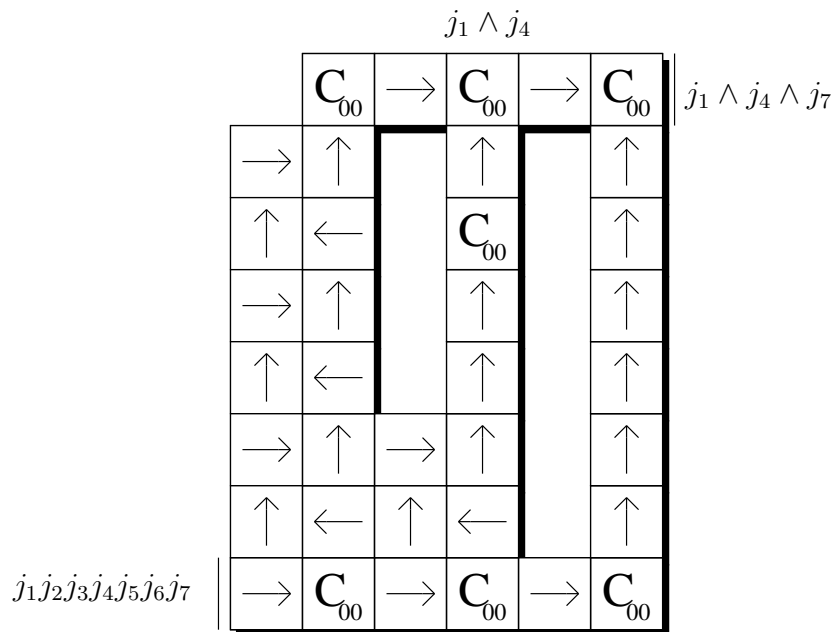


Figure 3.2: Décodeur D(1001001).

Remarquons finalement que D(1001001) décode la famille de séquences $1\phi\phi1\phi\phi1$, où ϕ dénote un 0 ou un 1.

3.3 La bascule SR

La figure 3.3 illustre une bascule SR. Cette dernière se compose de trois éléments principaux :

- ❶ un pulseur P(11111) commandé par le signal R;
- ❷ un second pulseur P(11111) commandé par le signal S;
- ❸ une boucle fermée de longueur cinq, implantant la bascule, constituée de trois états de transmission ordinaires et d'un confluent.

Etudions le fonctionnement du dispositif. Initialement, la sortie Q est évidemment égale à zéro. Lorsque le signal S passe à 1, le pulseur génère cinq excitations mémorisées dans la boucle fermée. Ainsi, à chaque coup d'horloge, la sortie Q est égale à 1. Le *reset* du système se révèle plus délicat. Lorsque R passe à 1, cinq excitations ordinaires sont générées par le pulseur. Elles sont ensuite transformées en excitations spéciales par l'état confluent repéré par le symbole *. La première détruit l'état confluent de la boucle. A cet instant, trois excitations ordinaires transitent encore dans la boucle. La reconstruction du confluent nécessite quatre coups d'horloge. Ainsi, au terme de ce processus, nous avons reconstitué la boucle. Cette dernière ne comportant plus aucune excitation, la sortie Q de la bascule est égale à zéro.

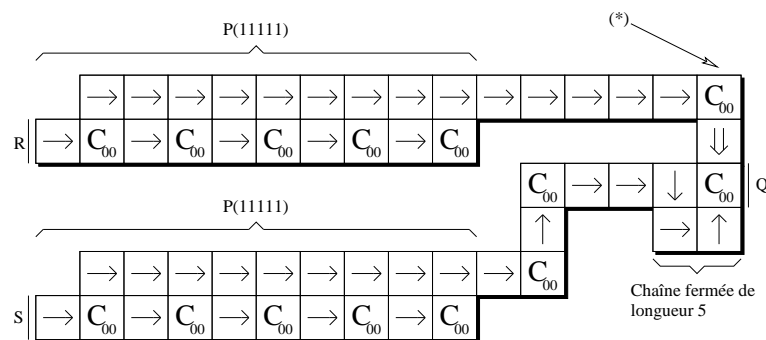


Figure 3.3: *La bascule SR.*

Chapitre 4

Réalisation matérielle

4.1 Introduction

Plusieurs chercheurs ont simulé, partiellement ou totalement, l'automate universel autoréplicateur de von Neumann :

- Jacqueline Signorini [4] a développé une machine SIMD spécialisée implantant la règle de transition; elle a ensuite simulé divers organes (pulseurs, décodeurs, ...).
- Umberto Pesavento [3] a proposé une extension de la règle de transition et simulé l'automate complet.

La première réalisation matérielle de l'automate de von Neumann date de 1995 [1,2]. Nous avons conçu un biodule représentant une cellule de ce dernier et en implantant la règle de transition. Un assemblage de plusieurs de ces circuits permet la construction d'organes simples (pulseurs et décodeurs)¹.

La solution consistant à implanter la règle de transition à l'aide d'une *look-up table*, faisant correspondre à chaque 5-uplet d'états un état futur, est illusoire. Elle nécessite en effet une table de recherche de $29^5 = 20'511'149$ entrées. Nous utilisons donc une architecture plus subtile que nous présentons dans ce chapitre.

4.2 Description du logidule

Les biodules “von Neumann” (602), dont le couvercle est reproduit sur la figure 4.1, ont une taille de 8x8 cm.

Afin de réaliser un mini-réseau cellulaire, il suffit de les connecter ensemble, comme indiqué sur la figure 4.2. Chacun des biodules possède les entrées/sorties suivantes :

¹Nous ne disposons évidemment pas de suffisamment de biodules pour concevoir le constructeur universel.

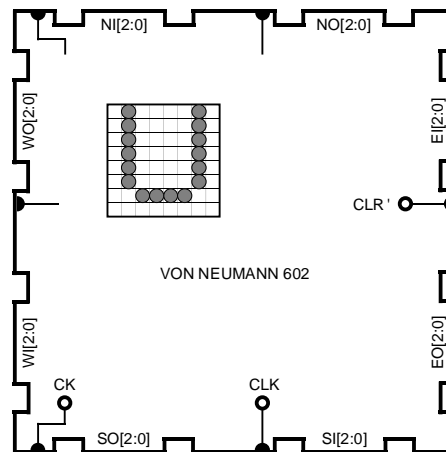


Figure 4.1: Couvercle du biodule “von Neumann”.

- 8 x 3 bits permettant la communication entre cellules;
- un signal de reset (CLR') mettant la cellule dans l'état U;
- un signal d'horloge CK rythmant l'évolution du réseau (fréquence typique : quelques Hertz);
- un signal d'horloge CLK utilisé pour l'affichage de l'état (fréquence minimale : 1 MHz).

Les connexions automatiques fournies par les boîtiers des biodules sont exploitées au maximum afin de réduire autant que possible le nombre de fils.

4.3 Architecture générale

L'architecture générale du circuit logique monté dans le biodule “von Neumann” est illustré par la figure 4.3. Le biodule “von Neumann” est composé de deux modules indépendants :

❶ la partie de calcul

- mémorise l'état courant de la cellule,
- communique directement avec les cellules voisines et
- calcule son état futur;

❷ le module d'affichage

- reçoit de la partie de calcul cinq bits codant l'état courant et
- s'occupe de le représenter graphiquement.

La partie de calcul est composée d'un système logique combinatoire et d'un ensemble de bascules mémorisant l'état courant de la cellule, comme indiqué sur la figure 4.4.

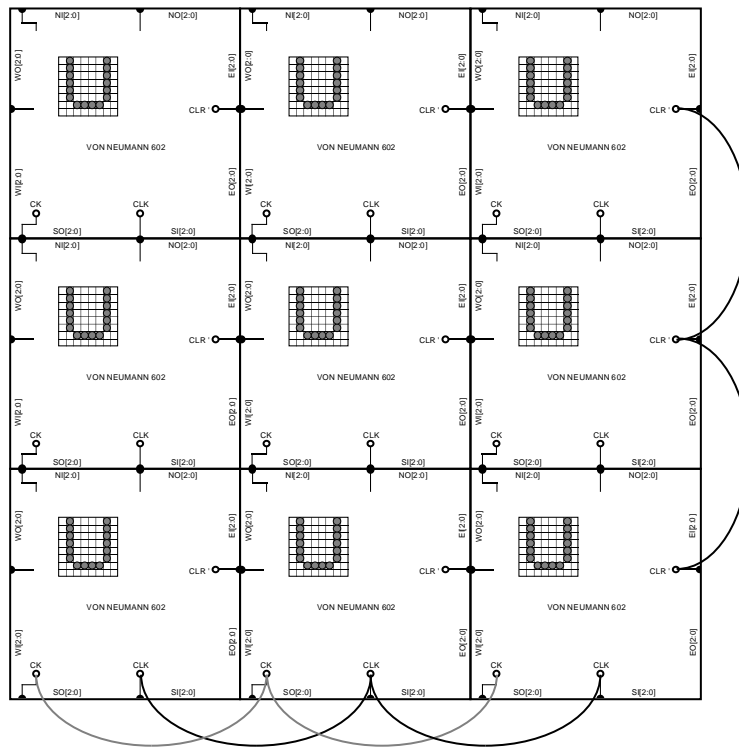


Figure 4.2: Connexion de plusieurs biodules.

4.4 Codage des états

Le choix de la stratégie de codage des états s'avère primordial. Parmi les solutions possibles, citons :

- un **codage minimal**
 - ⇒ capable de stocker l'état courant sur cinq bits
 - ⇒ avantage : taille réduite
 - ⇒ inconvénient : décodage complexe
- un **codage 1 parmi 29**
 - ⇒ nécessite 29 bits pour le stockage de l'état courant
 - ⇒ avantage : décodage très aisé
 - ⇒ inconvénient : taille relativement grande

Nous avons opté pour un codage sur 10 bits regroupant les avantages de chacun des deux codages présentés ci-dessus. Ces 10 bits sont séparés en trois champs distincts (figure 4.5) :

- ❶ Quatre bits indiquent le type de l'état (tableau 4.1). Les états sensitifs ont été arbitrairement numérotés de 0 à 7.

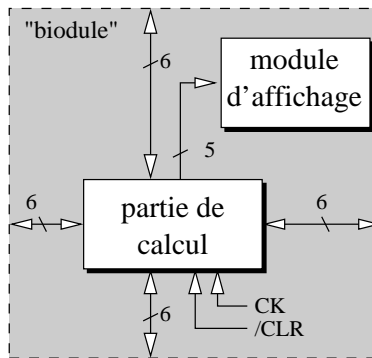


Figure 4.3: Schéma-bloc du biodule "von Neumann".

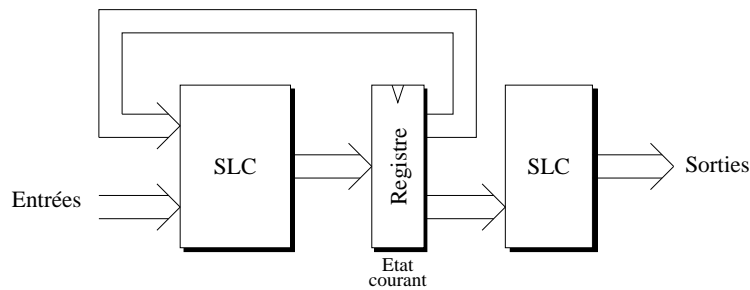


Figure 4.4: Schéma-bloc de la partie de calcul.

② Deux bits indiquent si une cellule est excitée ou non. Leur signification varie en fonction de l'état de la cellule :

- Etat confluent : ces deux bits indiquent les excitations courante et future.
- Etat de transmission (ordinaire ou spécial) : un bit suffit à la mémorisation de l'excitation, le second demeure inutilisé.
- Etats sensitifs et quiescent : ces bits sont inutilisés.

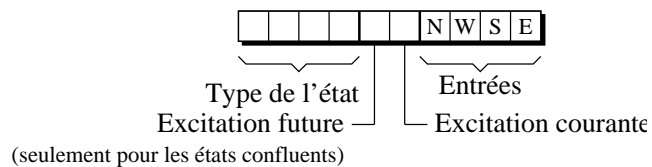


Figure 4.5: Codage sur 10 bits divisés en trois champs.

Code	Signification
0000	Etat quiescent
0001	Etat de transmission spécial
0010	Etat confluent
0011	Etat de transmission ordinaire
$1\phi\phi\phi$	Etat sensitif numéro $\phi\phi\phi$

Tableau 4.1: Codage du type de l'état.

- ④ A chacun des quatre côtés d'une cellule est associé un bit indiquant s'il s'agit d'une entrée. Ainsi, trois de ces bits sont égaux à 1 pour un état de transmission. Tous sont fixés à 1 pour un état confluent, ou à 0 pour les états sensitifs et quiescent.

Le tableau 4.2 présente quelques exemples de codes d'états.

Code	Etat
0011 00 0111	\uparrow
0010 10 1111	C_{01}
0001 01 1110	\Rightarrow
0000 00 0000	U

Tableau 4.2: Exemples de codes d'états.

4.5 Communication entre cellules

A priori, il semble nécessaire que chaque cellule connaisse l'état de ses quatre voisines. Nous pouvons cependant nous affranchir de cette contrainte. En effet, une cellule réagit de manière identique si, par exemple, ses voisines sont dans l'état quiescent ou de transmission spécial non excité.

La communication entre cellules s'effectue grâce à l'échange de messages. Nous avons déterminé un protocole comportant cinq messages distincts :

- ① “néant”, émis, par exemple, par une cellule dans l'état quiescent;
- ② “excitation ordinaire”, émis dans une seule direction par un état de transmission ordinaire excité;
- ③ “excitation spéciale”, émis dans une seule direction par un état de transmission spécial excité;
- ④ “excitation confluyente”, émis dans toutes les directions par une cellule dans l'état C_{10} ou C_{11} ;
- ⑤ “état de transmission ordinaire non excité”, émis dans une seule direction par un état de transmission ordinaire non excité.

Il suffit donc de disposer de trois bits dans chaque direction et dans chaque sens (entrée/sortie).

Code	Signification
$0\phi\phi$	néant
100	état de transmission ordinaire non excité
101	excitation spéciale
110	excitation confluyente
111	excitation ordinaire

Tableau 4.3: *Communication entre cellules.*

Le codage utilisé pour ces trois bits est indiqué dans le tableau 4.3.

4.6 Système logique

Le système logique réalisant la règle de transition de von Neumann a été implanté dans un circuit FPGA. L'architecture générale du circuit est donnée par la figure 4.6.

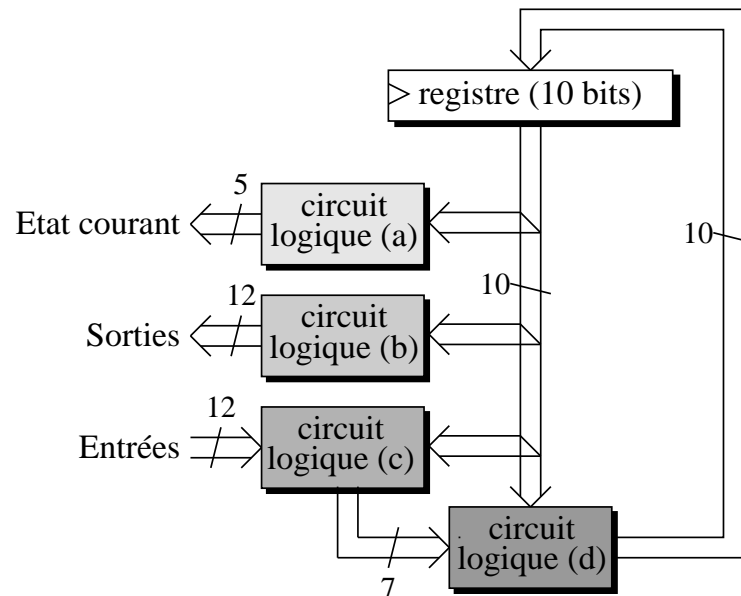


Figure 4.6: *Schéma-bloc du FPGA.*

Le registre 10 bits mémorise l'état courant de la cellule, alors que les autres blocs remplissent les fonctions suivantes :

- ❑ le circuit logique (a) traduit le code de l'état sur 10 bits en un code sur cinq bits (codage minimal) en vue de sa transmission au module d'affichage;
- ❑ le circuit logique (b) génère les 12 (= 3 bits · 4 directions) bits de sortie qui ne dépendent que de l'état courant;

- le circuit logique (c) fournit 7 signaux de contrôle au circuit logique (d) en fonction de l'état courant de la cellule et de la valeur des 12 bits d'entrée;
- le circuit logique (d) calcule l'état futur de la cellule en fonction de l'état courant de cette dernière et des 7 signaux de contrôle générés par le circuit logique (c).

Les 7 signaux de contrôle sortant du circuit logique (c) sont les suivants :

- ❶ **Excitation** : la cellule reçoit une excitation susceptible de l'exciter;
- ❷ **Confluent** : la cellule est dans un état confluent;
- ❸ **All Excited** :
 - au moins un état de transmission ordinaire excité est dirigé vers la cellule courante et
 - tous les états de transmission ordinaires dirigés vers la cellule courante sont excités;
- ❹ **Sens Excitation** : la cellule courante reçoit une excitation ordinaire ou spéciale;
- ❺ **U** : la cellule se trouve dans l'état quiescent;
- ❻ **S** : la cellule se trouve dans un état sensitif;
- ❼ **Kill** : la cellule doit être détruite au coup d'horloge suivant.

Bien que relativement simples, les équations de ces signaux ne sont pas présentées ici. Le lecteur intéressé se référera au rapport technique décrivant les biodules [1] pour plus de détails.

A l'aide de ces 7 signaux et de l'état courant de la cellule, il est aisé de calculer son état futur. Le circuit logique (d) réalisant cette fonction est reproduit sur la figure 4.7.

L'évolution des états sensitifs a nécessité une gestion particulière, impliquant la mise en œuvre d'une table de recherche (*look-up table*).

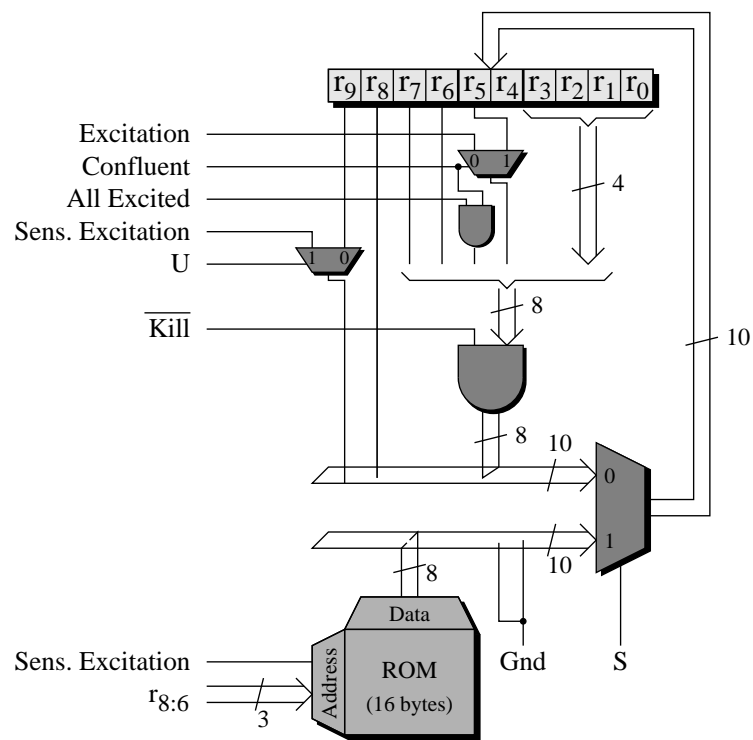


Figure 4.7: Schéma logique calculant l'état futur de la cellule.

Chapitre 5

Construction d'un organe

5.1 Description du problème

Lors du précédent chapitre, nous avons décrit des biodules implantant la règle de transition de von Neumann. L'assemblage de ces derniers permet de réaliser un réseau cellulaire.

Il reste toutefois encore un problème à résoudre. Comment implanter sur ce réseau un organe ? Autrement dit, comment assigner un état donné à une cellule particulière ?

La solution adoptée consiste à mettre en œuvre un bras constructeur. Ce dernier croît jusqu'à une cellule (dans l'état quiescent), puis transmet la séquence d'excitations permettant l'élaboration de l'état souhaité. Ce bras s'étend du constructeur jusqu'à une cellule particulière.

A l'enclenchement de nos biodules, toutes les cellules sont dans l'état quiescent. De plus, nous ne disposons pas de constructeur universel. Nous utilisons par conséquent un "constructeur" extérieur au système.

Dans son ouvrage, von Neumann décrit deux types de bras constructeur différents. Ils font l'objet des deux paragraphes suivants.

5.2 Bras simple

La première solution consiste à réaliser un bras formé d'une succession d'états de transmission comme indiqué sur la figure 5.1. Extensible à volonté, ce bras croît jusqu'à une cellule particulière, puis achemine les excitations nécessaires à la construction de l'état souhaité.

Une fois cette cellule construite, il n'est pas possible d'avancer le bras plus avant, sous peine d'écraser l'état nouvellement construit. De plus, il n'est pas possible de "reculer" le bras.

Le procédé permettant la construction successive de plusieurs cellules adjacentes consiste à écraser le bras précédemment conçu. Pour cela, nous créons alternativement des bras formés d'états de transmission ordinaires et spéciaux.

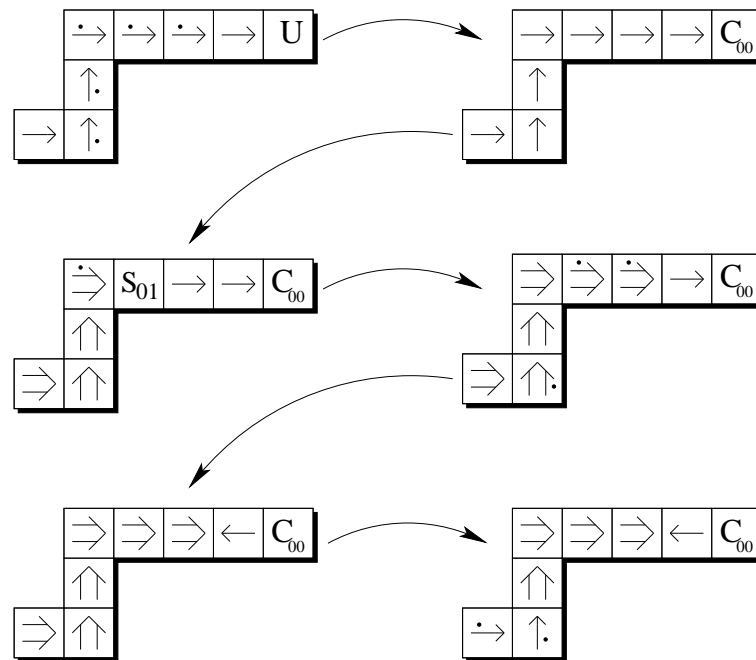


Figure 5.1: *Construction de cellules à l'aide d'un bras simple.*

5.3 Bras double

L'impossibilité de reculer le bras simple d'une cellule implique qu'il faut le reconstruire pour chaque cellule à créer. Ce processus se révèle coûteux.

Von Neumann a proposé un autre type de bras constructeur, que nous appellerons "bras double". Il s'agit du bras utilisé dans la version finale de l'automate universel auto-reproducteur.

Ce bras est formé d'un double alignement d'états de transmission ordinaires, respectivement spéciaux. Ce bras est capable

- d'avancer d'une cellule vers la droite,
- d'avancer d'une cellule vers le haut,
- de construire deux cellules adjacentes,
- de reculer d'une cellule.

Le principe de construction de deux cellules est explicité par la figure 5.2. On peut y voir le bras reculer d'une cellule et construire deux états (nommés A et B) sur les deux cellules ainsi libérées.

La figure 5.3 montre comment il est possible d'avancer le bras d'une cellule.

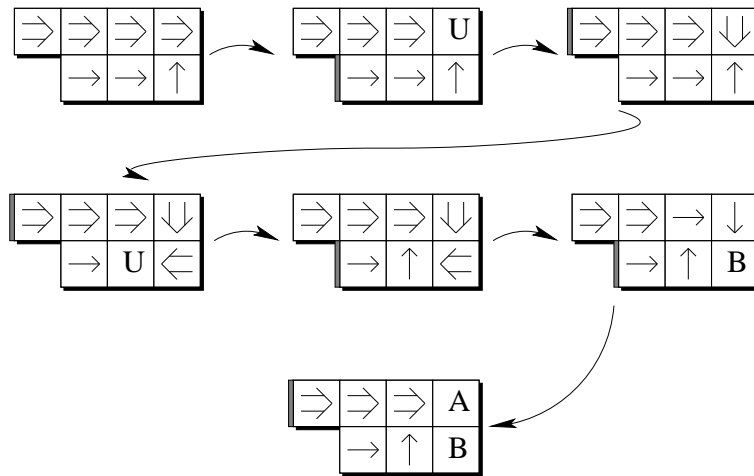


Figure 5.2: Construction de deux cellules à l'aide d'un bras double.

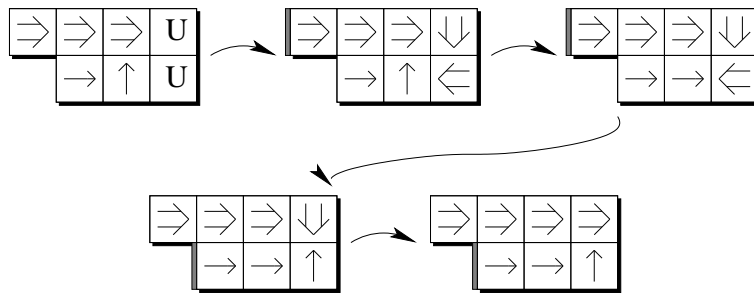


Figure 5.3: Avance du bras d'une cellule.

5.4 Solution retenue

Le bras simple prend peu de place sur le réseau, mais nécessite un temps de construction important. Le bras double permet des constructions plus rapides, mais requiert plus de place (2 cellules de large).

Certains organes contiennent des cellules dans l'état quiescent. Plutôt que de mettre des biodules pour représenter ces cellules, nous préférons laisser des trous dans le montage. Ceci nous empêche d'utiliser le bras double, par manque de place sur le réseau.

Par conséquent, nous allons mettre en œuvre un bras de construction simple, qui sera contrôlé par un circuit externe (Macintosh ou carte spécialisée).

Chapitre 6

Démonstrations

Nous décrivons ici les expériences présentées dans la vitrine consacrée à von Neumann au Laboratoire de Systèmes Logiques. Une petite carte comportant un microcontrôleur et de la mémoire fournit les signaux à un réseau de biodules 602. Les descriptions de deux automates sont stockées en mémoire :

- Le premier automate (figure 6.1), un pulseur P(101), construit un état de transmission ordinaire. Cet état est ensuite détruit par une excitation spéciale.
- Le deuxième (figure 6.2) est un pulseur P(11001) créant, puis détruisant, un état de transmission spécial.

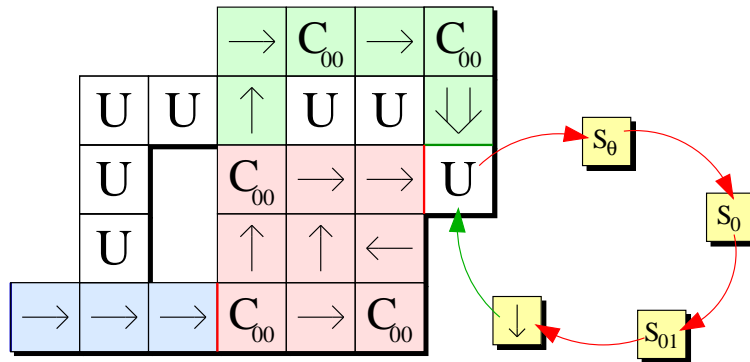


Figure 6.1: *Exemple d'organe : pulseur P(101), avec destruction d'une cellule.*

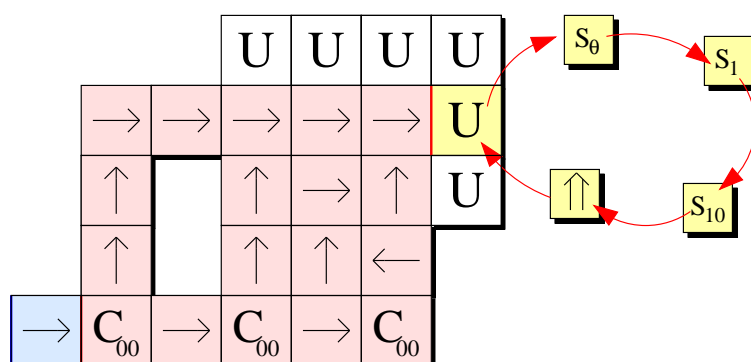


Figure 6.2: Exemple d'organe : pulseur $P(11001)$.

Annexe A

Exercices

Nous proposons quatre exercices permettant au lecteur de se familiariser avec l'automate cellulaire de von Neumann. Les solutions sont présentées dans l'annexe B.

EXERCICE 1

Réaliser un pulseur $P(1001001)$.

EXERCICE 2

Réaliser un pulseur $P(11101)$.

EXERCICE 3

Réaliser un décodeur $D(101001)$.

EXERCICE 4

Concevoir un automate calculant le "ou exclusif" de deux variables booléennes A et B .

Annexe B

Solution des exercices

B.1 Pulseur P(1001001)

La première étape consiste à déterminer la taille de l'automate. Rappelons que pour un pulseur, la hauteur h et la largeur w sont respectivement de $2 \cdot k$ et $u + 2$ cellules, où u et k dénotent le nombre de 0 et de 1 de la séquence. Dans cet exemple, $u = 4$ et $k = 3$. Par conséquent, $w = 6$ et $h = 6$. La figure B.1 illustre l'automate réalisant la tâche souhaitée.

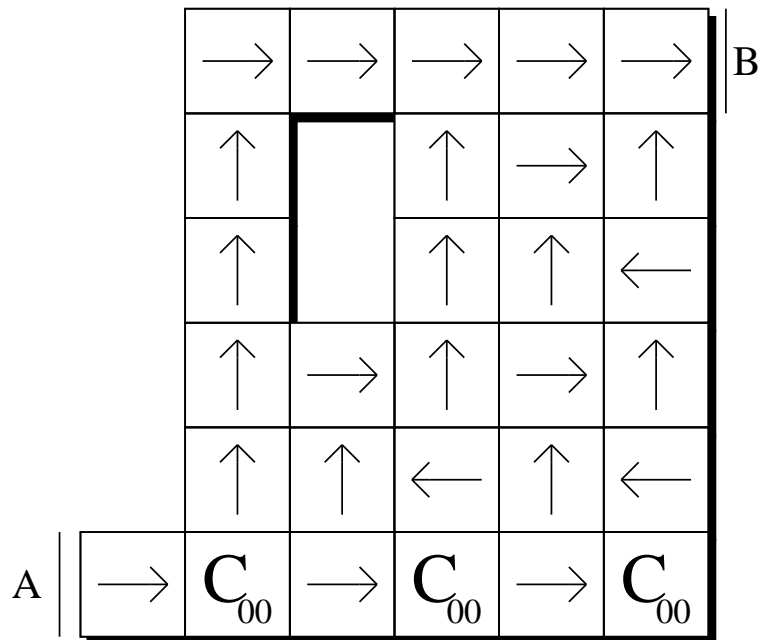


Figure B.1: *Pulseur P(1001001)*.

B.2 Pulseur P(11101)

Ce pulseur se conçoit de la même manière que celui du premier exercice. Comme la séquence ne comporte qu'un seul 0, $u = 2$. La figure B.2 illustre le dispositif.

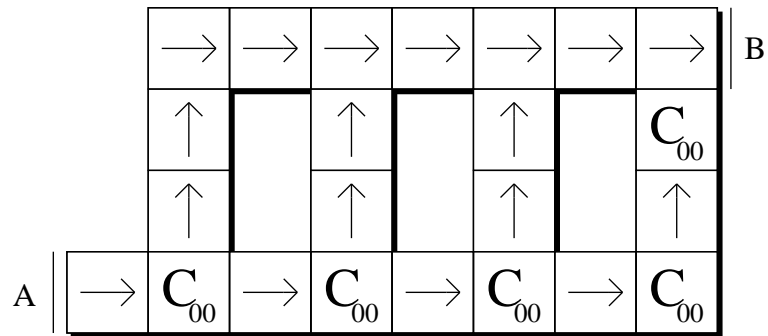


Figure B.2: Pulseur P(11101).

B.3 Décodeur D(101001)

Soient k le nombre de 1 apparaissant dans la séquence et n la taille de cette dernière. La largeur et la hauteur du décodeur sont respectivement de $2 \cdot k = 6$ et $n + 2 = 8$ cellules. La figure B.3 illustre l'automate. Remarquons que ce dernier décode les séquences 101001, 101011, 101101, 101111, 111001, 111011, 111101 et 111111.

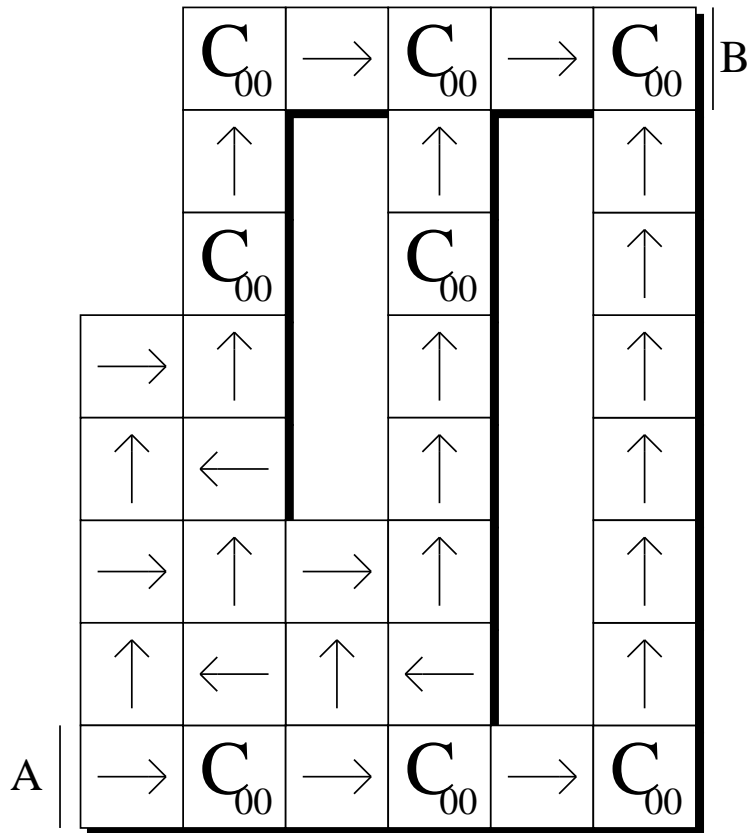


Figure B.3: Décodeur $D(101001)$.

B.4 Ou exclusif de deux variables

La figure B.4 illustre une solution possible à ce problème. Lorsque les deux entrées sont égales à 0, il est évident que la sortie est égale à 0. Etudions maintenant les autres situations.

❶ $A = 1$ et $B = 1$

La cellule C_{00} repérée par le symbole * reçoit deux excitations ordinaires. Elle fournit par conséquent une excitation au pulseur $P(110000)$ (en gris sur la figure). Au temps $t=14$, la cellule noire reçoit deux excitations ordinaires ainsi qu'une excitation spéciale. La cellule est par conséquent détruite et la sortie du système est égale à zéro. Au coup d'horloge suivant, le pulseur fournit un 1. La cellule noire passe donc dans l'état S_θ . Lors des quatre prochains coups d'horloge, elle ne reçoit plus aucune excitation. Au terme de ce processus de construction, nous obtenons un état de transmission ordinaire orienté vers la droite. La porte "ou exclusif" est prête pour un nouveau calcul.

❷ $A = 1$ et $B = 0$

Le pulseur $P(110000)$ n'étant pas activé, la cellule noire ne reçoit qu'une

excitation ordinaire au temps $t = 14$. La sortie du dispositif est ainsi égale à 1.

③ $A = 0$ et $B = 1$

Cette situation est similaire à celle décrite au point 2.

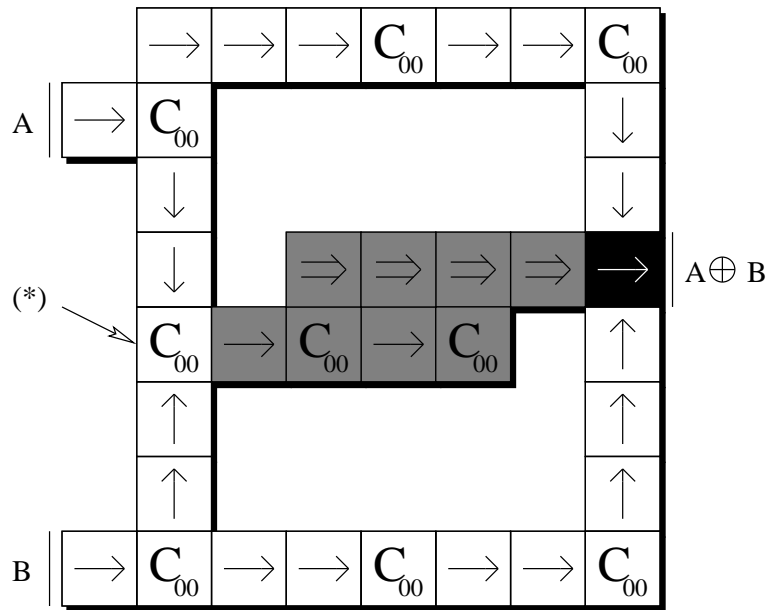


Figure B.4: Ou exclusif de deux variables. Le pulseur $P(110000)$ apparaît en gris. La sortie du système est dessinée en noir.

Bibliographie

- [1] Jean-Luc Beuchat and Jacques-Olivier Haenni. Logidule de von Neumann. Technical report, Logic Systems Laboratory, IN – Ecublens, 1015 Lausanne, Switzerland, June 1995.
- [2] Jean-Luc Beuchat and Jacques-Olivier Haenni. Von Neumann’s 29-State Cellular Automaton: A Hardware Implementation. *IEEE Transactions on Education*, 43(3):300–308, 2000.
- [3] Umberto Pesavento. Von Neumann’s Universal Constructor, September 1995. submitted.
- [4] Jacqueline Signorini. How a SIMD machine can implement a complex cellular automaton ? A case study: von Neumann’s 29-state cellular automaton. In *ACM and IEEE Proceedings of the Supercomputing '89*, pages 175–186, 1989.
- [5] John von Neumann. *Theory of Self-Reproducing Automata*. University of Illinois Press, 1966. edited by A. W. Burks.

Table des matières

1	Prolégomènes	1
1.1	Brève biographie de John von Neumann	1
1.2	Principes fondamentaux	1
2	Description des états et de la règle de transition	5
2.1	L'état quiescent	5
2.2	Les états de transmission ordinaires	5
2.3	Les états de transmission spéciaux	6
2.4	Les états confluents	6
2.5	Les états sensitifs	8
2.6	Destruction d'une cellule	9
3	Applications	11
3.1	Le pulseur	11
3.2	Le décodeur	12
3.3	La bascule SR	13
4	Réalisation matérielle	15
4.1	Introduction	15
4.2	Description du logidule	15
4.3	Architecture générale	16
4.4	Codage des états	17
4.5	Communication entre cellules	19
4.6	Système logique	20
5	Construction d'un organe	23
5.1	Description du problème	23
5.2	Bras simple	23
5.3	Bras double	24
5.4	Solution retenue	25
6	Démonstrations	27
A	Exercices	29

B	Solution des exercices	31
B.1	Pulseur P(1001001)	31
B.2	Pulseur P(11101)	32
B.3	Décodeur D(101001)	32
B.4	Ou exclusif de deux variables	33